OpenFlow環境におけるPacket-Inメッセージの特性に基づくポートスキャン検出手法に関する研究

修士学位論文要約（令和 3 年 3 月）

OpenFlow 環境における Packet-In メッセージの特性に基づくポートスキャン検出手法に関する研究

小野 大地

指導教員：菅沼 拓夫

A Study on a Port Scan Detection Method Based on Packet-In Messages in OpenFlow Networks

Daichi ONO

Supervisor: Takuo SUGANUMA

OpenFlow (OF) is the de-facto standard that enabled the implementation of Software-Defined Networking (SDN). However, OF can be vulnerable under Port scans as a recognizance method; thus, it is essential to detect it before the full-scale attack. Conventional approaches periodically execute a detection process, which increases the overhead significantly. In this paper, we propose a method that considers the characteristics of Packet-In messages in OF to promptly and efficiently detect port-scans. We tested the proposed approach using simulated and real traffic data, and the results show a considerable improvement in the detection time and much lower overhead than existing methods.

1. 序論

ネットワークをソフトウェアで柔軟に制御・管理する仕組みである Software Defined Network (SDN) が注目され，その実装である OpenFlow が普及しつつある．また，SDN/OpenFlow 環境に関して，サイバー攻撃の準備段階として行われるポートスキャンを検出する試みがある．既存研究 [1] では，短い間隔で全ての OpenFlow スイッチから統計情報を OpenFlow コントローラが収集し，検出処理を行っているため，コントローラのオーバーヘッドが増加してしまうという課題があった．そこで本研究では，スイッチが持つ統計情報だけでは無く，スイッチからコントローラに送られる Packet-In メッセージの特徴を考慮することで，より効率的にポートスキャンを検出する手法を提案する．

2. 関連技術・研究

2.1 OpenFlow

OpenFlow スイッチは，どのパケットをどのように処理するかを示したフローエントリと呼ばれる情報を持つ．また，各フローエントリ毎に統計情報を記録しており，受信パケット数やフローエントリが作られてからの経過時間等を OpenFlow コントローラで収集することが可能である．スイッチはパケットを受信したときに自身が持つフローエントリにマッチするものが存在しない場合に，コントローラに対して Packet-In メッセージを送信することで処理方法の問い合わせを行うことが可能である．

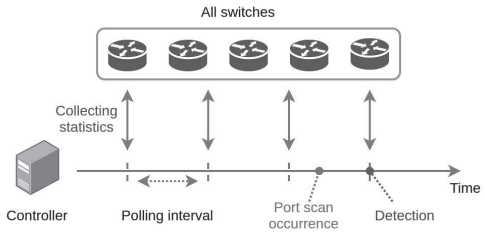

図 1: ポーリング型手法のイメージ図

2.2 OpenFlow 環境におけるポートスキャン検出

OpenFlow 環境におけるポートスキャン検出に関する既存研究 [1] では，図 1 のように，各スイッチの持つフローエントリの統計情報を数秒間隔で周期的に収集・分析することでポートスキャンの有無を判定する．この手法では，短い間隔で何度も検出処理が発生するため，ポートスキャンの発生頻度が低い場合に不要な処理が多発してしまう．また，全てのスイッチから統計情報を収集しているため，ネットワークを構成するスイッチ数の増加に応じてオーバーヘッドが増加してしまうという課題がある．

3. 提案

本研究では，前章で述べた課題に対する新たなポートスキャンの検出手法の検討を行う．一般にポートスキャンが発生すると，短時間で複数のポートに対するパケットが発生する．OpenFlow 環境においては，スイッチが初めて受信する種類のパケットに対して Packet-In メッセージが発生させることが可能が

あることから，ポートスキャンが発生した場合にコントローラに大量の Packet-In メッセージが送られる．本研究では，この性質に着目した手法を提案する．提案手法のイメージを図 2 に示す．提案手法では，各スイッチからコントローラに対して送られる Packet-In メッセージの流量を，送信元となったホストの Ethernet アドレス毎にモニタリングし，流量の異常増加を起点に特定のスイッチからのみ統計情報を収集し，ポートスキャンの検出処理を行う．この手法によって，全てのスイッチから周期的に統計情報を収集する必要が無くなるため，コントローラのオーバーヘッド低減が期待できる．また，Packet-In メッセージの異常増加検出手法には Spectral Residual 法 (SR)[2] を用いた．SR は，非定常なデータに対しても適用が可能であり，手動によるラベル付けが不要で，高速にオンライン処理が可能であるという利点を持ち，提案手法に非常に適していると考えられる．

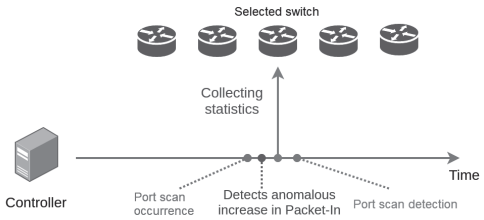

図 2: 提案手法のイメージ図

4.　実験

提案手法の有効性を評価するために，周期的に全てのスイッチから統計情報を収集・分析する手法 (ポーリング型手法) との CPU オーバーヘッドの比較実験を行った．実験のために，Ryu コントローラと Open vSwitch を用いて仮想的な OpenFlow ネットワークを構築した．実験では，2 ～ 10 台のスイッチが直列に接続され，各スイッチがそれぞれ 3 台のホストを持つトポロジーを用意し，各ホストが正常なトラフィックを送出し続けている間にポートスキャンを行った際のコントローラの平均 CPU 使用率を測定した．実験結果として，ネットワークの規模に対するコントローラの平均 CPU 使用率を図 3 に示す．

図から，提案手法はポーリング型手法と比較してどのスイッチ数においても平均 CPU 使用率を低く抑えることが出来ていることが分かる．また，提案手法のグラフはポーリング型手法のものよりも勾配が小さいことから，ネットワークの規模に対してよりスケーラビリティに優れていることが分かる．

加えて，ポートスキャン発生時の Packet-In メッセージの異常増加について，提案手法による検出精度の評価実験を行った．実験では，研究室で収集した平日 5 日間の正常なトラフィックデータと，Nmap と呼ばれる代表的なポートスキャンツールで発生さ

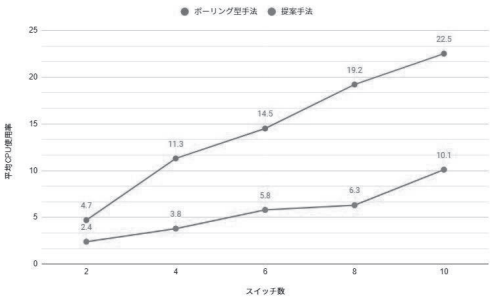

図 3: ネットワークの規模に対するコントローラの平均 CPU 使用率

せたポートスキャンのトラフィックデータを用いて提案手法の検出精度を評価した．Nmap には，T0 ～ T5 までのプローブ送信のタイミングテンプレートが用意されており，数字が高くなればなるほど高レートとなる．実験の結果を表 1 に示す．

表より，T2 以上のレートであればポートスキャンに起因する Packet-In メッセージの異常増加を十分な精度で検出可能であることが分かる．

表 1: Nmap の各タイミングテンプレートにおける提案手法の精度

テンプレート	Accuracy	Precision	Recall	FPR
T0 (Paranoid)	0.784	0.285	0.190	0.096
T1 (Sneaky)	0.820	0.457	0.380	0.091
T2 (Polite)	0.968	0.947	0.857	0.009
T3 (Normal)	1.0	1.0	1.0	0.0
T4 (Aggressive)	1.0	1.0	1.0	0.0
T5 (Insane)	1.0	1.0	1.0	0.0

5.　結論

本研究では，OpenFlow 環境におけるポートスキャン検出において，Packet-In メッセージの性質を考慮することでより効率的に検出を行う手法を提案した．また，実験から，提案手法はポーリング型手法よりも低いオーバーヘッドかつ高いスケーラビリティを実現できることを示した．検出精度については，T3 以上のレートでは十分であったものの，T1 以下については不十分であったため，今後の課題として，ポーリング間隔を長く取ったポーリング型手法との併用を検討する必要があると考えられる．

参考文献

1) Neu CV et al.: ”Lightweight IPS for port scan in OpenFlow SDN networks,” Proc. of NOMS (2018), pp. 1-6. 2018
2) Ren H et al.: ”Time-Series Anomaly Detection Service at Microsoft,” Proc. of KDD’19 (2019)